Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 44 (2015) 244 – 253
Procedia Computer Science

2015 Conference on Systems Engineering Research

Knowledge Based Development in Automotive Industry guided by Lean Enablers for System Engineering

Daniel Stenholm[a]*, Henrik Mathiesen[b], Dag Bergsjo[c]

[a]*Chalmers University of Technology, Horsalsvagen 7A, Gothenburg SE-412 96, Gothenburg*
[b]*Kongsberg Automotive, Kongsberg, Norway*
[c]*Buskerud and Vestfold University College, Kongsberg, Norway*

Abstract

The Systems Engineering literature acknowledges that the principles of Lean Product Development foster the achievement of higher program performance[5]. Work has been carried out to explore and capture synergies between traditional System Engineering, Program Management and Lean, which has been described as Lean Enablers for Managing Engineering Programs. Kongsberg Automotive in Kongsberg, Norway has transformed from a traditional product development process to a new knowledge focused process called Knowledge Based Development over a period of three years. This paper addresses the transformation from both a theoretical and practical perspective and maps the supporting tools that constituted a large part of the change. Additionally, the paper addresses the Lean Enablers (LE) that have been affected due to obstacles in transformation and future potential LE which have not yet been reached.

Peer-review under responsibility of the Stevens Institute of Technology.

Keywords: Knowledge Based Development; Lean Enablers for System Engineering; Lean Product Development.

1. Introduction

Knowledge Based Development (KBD) presents ways for companies to restructure and improve their organizations, and a central focus is that knowledge and learning are critical in system engineering. The goal with implementing KBD at Kongsberg Automotive (KA) was to decrease repetitive problems and time to market as well

* Corresponding author. Tel.: +46-739-077-636.
E-mail address: Daniel.Stenholm@chalmers.se

Peer-review under responsibility of the Stevens Institute of Technology.
doi:10.1016/j.procs.2015.03.047

as increase quality by reusing knowledge to a greater extent than today. In this paper, four tools and methods are described to support the knowledge value stream (KVS); LAMDA as a culture, A3 for problem-solving, Trade-off curves for visualizing feasible design areas and Checksheets to support knowledge reuse and decision making. To further validate the tools and methods as valuable and find improvement areas, Lean Enablers (LE) for managing engineering programs were mapped and then used as complementary questions during interviews. KA is presented as a case company that adapted the KBD in their system engineering process. They are not yet satisfied but have made progress and together with this study, consisting of 13 interviews and observation carried out over 6 months, a plan for future activities is set. The new process has its origin in lean product development and knowledge management principles.

The overall question addressed by this paper is *how the new KBD process has been implemented at KA and what major challenges and opportunities still exist having reflected over the KBD rollout*.

The paper is structured as follows; first knowledge regarding KBD, System Engineering (SE) and Lean SE is presented. This is followed by mapping between KBD tools and methods to System Engineering with support from LEs, and then a description of the case company. The rollout of the KBD process including goals, initiatives connected to the process as well as expected and realized results are described. Finally an evaluation of implemented methods and tools and remaining challenges concerning the rollout, in particular the parts that involve knowledge creation, storage and reuse are discussed. The paper concludes with discussion and conclusion chapters

2. Knowledge Based Development and Systems Engineering

KBD is derived from general systems engineering methodology but particularly from lean product development and the notion of the knowledge value stream (Fig. 1), presented by Kennedy[1]. The idea is to generate useful knowledge about both current product/project but also incorporate a process of continuous learning in the knowledge value stream to create and reuse knowledge over time. To support this flow of knowledge both organizational systems and different tools (and potentially IT-systems) must be synchronized and harmonized throughout the organization. Even though it is not always apparent, quality problems are frequently repeated within and between projects. The term “reinventing the wheel” is commonplace, yet it exhibits the essence of the disregard for knowledge management (KM)[10].

The SE process has an iterative nature that supports learning and continuous improvement. As the processes unfold, systems engineers uncover the real requirements and the emergent properties of the system. Complexity can lead to unexpected and unpredictable behavior of systems; therefore, one of the objectives is to minimize undesirable consequences. This may be accomplished through the inclusion of and contributions from experts across relevant disciplines coordinated by the system engineers[5].

2.1. Tools and Methods in Knowledge-Based Development

There are four different tools and methods that are significant when talking about KBD; the LAMDA learning process to create knowledge, A3 reports for simple and purposeful communication, Limit & Trade-off curves (further mentioned as Trade-off curves) for visible representation of performance limits and Checksheets as a basis for reviews, owned and/or controlled knowledge[1].

Adapting **LAMDA** (Look – Go see for yourself, Ask – Get to the root cause of the problem, Model – Use some kind of analysis simulation or prototypes, Discuss – Communicate with mentors, developers of interfacing subsystems, etc., Act – Test your understanding experimentally) as a company culture and mindset supports and drives the company to act on knowledge in front of gut feeling. LAMDA is traditionally seen as a learning cycle for problem solving and is here adapted as a culture where it should be a mindset in every situation. It emphasizes knowledge creation, and true understanding of the root cause prior to acting on what first comes to mind[1].

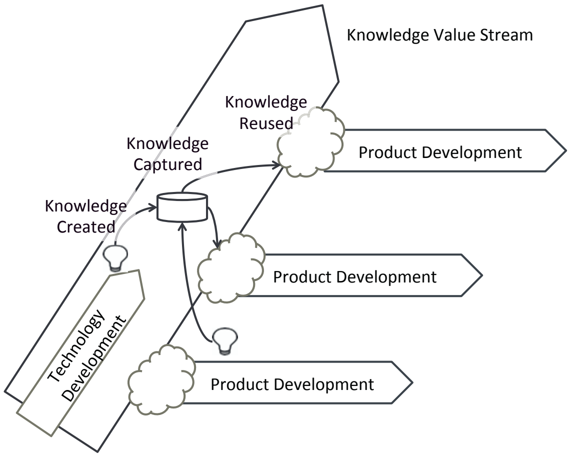

Fig. 1. Knowledge Value Stream adapted from Kennedy [1]

A known tool in the lean product development process is the **A3-reports**, which originally refer to Toyotas form to communicate complex information and solve problems. These are created on a single sheet of paper[3]. The name "A3" originates from a paper size (297 × 420 mm), which seems to be a good size to limit the report space available to the creator. When the A3 report is done it is usually stored digitally on the company server. One characteristic of A3s is the standardized form that makes it easier to read [1, 3, 8, 9].

To increase the understanding and enable thorough information in spite of its compact form, visual information is recommended in the largest possible degree. The size limit fosters well-defined descriptions of one concentrated subject, which can be positive but also negative in that multiple A3s may be created to describe different aspects of a subject, resulting in an increased number of reports. In the LPD literature different types and purposes for A3-reports are suggested[3, 9], although the focus in this paper is on Problem Solving A3s, which are the most common type.

Problem Solving A3s encourage systematic problem solving, including problem formulation and experimental design, which addresses high quality solutions to immediate local problems. Important to remember is that if a problem is small enough and local enough, it might not even need an A3. However, most problems benefit from the added rigor that writing a Problem Solving A3 provides [7].

To improve knowledge reuse and to present information in a visual way, **trade-off curves** have shown great benefits. A trade-off curve is a graph that shows one performance criterion on the Y-axis, and another performance criterion on the X-axis. A curve is plotted to illustrate the relation between the two criteria to predict the performance[3]. In this way, different design alternatives are considered simultaneously as the curve represents the collected characteristics of all solutions of a subsystem. If the desired point on the plot is within the feasible region, e.g. above the curve, it is safe to assume that the design can cope with the requirements. One of the first things that need to be established is which parameters affect each other, to be able to adapt them to the two axes. This will help to reuse knowledge gained about the limits of the design to increase the product performance.

If a design falls into the "unsafe" region of a curve, that red flag helps to understand the risk involved. Most of the time, the engineers want to keep the designs well within the safe regions of the curve. Sometimes it is worth the risk, and then it is necessary to learn how to transcend the limits of the current system as it is now understood. Either way, active use of these curves helps eliminate the major root cause of late design loopbacks: unanticipated problems in development that could have been avoided if the developer had access to the organization's knowledge.

Checksheets is a tool to gather knowledge. It can be seen as a collection of the existing knowledge in a certain area. Also referred to as engineering checklists, they are simple reminders of things that should not be left out when designing[3]. Checksheets is the basis for standardizing knowledge. They are used for validating the designs and are continuously updated[2]. Checksheets is ideally an accumulated knowledge base reflecting what has been learnt over time about both good and bad design practices, manufacturing requirements, critical to quality characteristics, critical design interface, performance requirements as well as standards that communize design[3].

Table 1 Knowledge-Based development tools and the main knowledge aspects.

	Summary	Main knowledge aspects
LAMDA	The process of Look, Ask, Model, Discuss, Act.	Is a learning process to e.g. gain more insights into a problem
A3s	Document, visualize and communicate on a single sheet of paper.	Documentation of learning and a problem solving method (combined with e.g. Lambda)
Trade-off Curves	A curve showing, according to the companies' best practices, feasible designs.	Manage best-known solutions and thereby make it possible to reuse knowledge in future project.
Checksheets	Summarizing design best practice in text together with links etc. Supporting tool for the designer to make the right decisions and also be able to go back and review decisions.	Standardize knowledge, focusing on What Why and How.

2.2. Lean in Systems Engineering

The field of Lean Systems Engineering (LSE) is "the application of Lean principles, practices, and tools of SE and to the related aspects of enterprise management in order to enhance the delivery of value while reducing waste"[6]. The goal is to deliver best lifecycle value for technically complex systems with minimum resources. And the value is defined as flawless mission assurance or product success delivered without waste, in fastest possible time.

Achieving excellence in system engineering programs is considered important but is highly challenging. Lean Enablers (43 Lean Enablers and 286 subenablers all referred as LE) set out to reflect on 10 main challenges affecting systems engineering program management and have been guided by the Lean Thinking philosophy. These challenges are, according to the book Lean Enablers for Managing Engineering Programs[5]: (i) firefighting -reactive program execution, (ii) unstable, unclear, and incomplete requirements, (iii) insufficient alignment and coordination of the extended enterprise, (iv) processes are locally optimized and not integrated for the entire enterprise, (v) unclear roles, responsibilities, and accountability, (vi) mismanagement of program culture, team competency, and knowledge, (vii) insufficient program planning, (viii) improper metrics, metric systems, and KPIs, (ix) lack of proactive program risk management and (x) poor program acquisition and contracting practices. The LEs are basically condensed "good sense" of actionable best practices for managing engineering programs. Implementation in systems engineering programs lay the basics for achieving the lean benefits and ultimately the program's excellence[5]. It is advised to start small by selecting the most beneficial Lean Enablers for the current program[5]. In this study, the most interesting LEs are those that link to the presented methods and tools for KBD. Lean Systems Engineering can, as all management approaches, be implemented through the adoption of tools, methods and best practices[6].

3. Mapping Knowledge Based Development tools and methods with Lean Enablers

LEs describe the synergies between traditional SE, Program Management and Lean. By mapping LAMDA, A3, Trade-off and Checksheets with relevant LEs the synergies between SE and KBD are presented. The LEs are then used to identify the outcome of the KBD process and whether or not the tools and methods are effectively implemented. A focus group along with literature performed the process of mapping the LEs to each method and tool.

Table 2 shows the mapping of Lean Enablers together with the tools and methods. This mapping can work as a set of LEs that can be measured to track and ensure the implementation of the new process to desired outcomes.

In addition, a parallel track to focus on the LE was carried out, Observations and 13 interviews were performed; six of them four and a half years after the start of KBD implementation and the remaining five years after, to further support and identify the pros and cons. The interviewees were mainly engineers with a variety of experience and departments, such as testing, concept development and project engineering.

Several people who work at KA were asked to comment on how the process change has affected their work in relation to the LEs in table 2. This was carried out as a part of the evaluation on how the KBD rollout has changed the organization. The result of the interviews and observations is found in the analysis chapter.

Table 2. LE mapped to KBD tools and methods. Numbers referring to Lean Enablers to Manage Engineering Programs[5]

#LE	Lean Enabler for Managing Engineering Programs	CS	A3	LAMDA	Trade-off
1.	**Lean Enablers to Treat People as Your Most Important Asset (Lean Principle 6)**				
1.1.07.	Build a culture of mutual trust and support (there is no shame in asking for help).			X	
1.1.10.	When resolving issues, attack the problem, not the people.		X	X	
1.3.	**Support an autonomous working style.**	X			
1.3.3.	Allow certain amount of "failure" in a controlled environment at lower levels, so people can take risk and grow by experience.			X	
1.3.5.	Keep management decisions crystal clear while also empowering and rewarding the bottom-up culture of continuous improvement and human creativity and entrepreneurship.			X	
1.4.1.	Establish and support communities of practice.	X			
1.4.6.	Establish a highly experienced core group ("gray hairs") that leads by example and institutionalizes positive behavior.	X			
1.5.	**Promote the ability to rapidly learn and continuously improve.**	X			
1.5.4.	Capture and share tacit knowledge to stabilize the program when team members change.	X			
1.6.1.	Prefer physical team co-location to the virtual co-location.			X	
1.6.3.	Promote direct human communication to build personal relationships.			X	
1.6.7.	Encourage (and document when appropriate) open information sharing within the program.	X	X		
2.	**Lean Enablers to Maximize Program Value (Lean Principle 1)**				
2.1.3.	Develop a robust process to capture, develop, and disseminate customer stakeholder value with extreme clarity.	X			
2.3.06.	Create shared understanding of program content, goals, status, and challenges among key stakeholders.	X			
2.3.09.	Listen to the stakeholders` comments and concerns patiently and value their views and inputs.			X	
2.3.10.	Clearly track assumptions and environmental conditions that influence stakeholder requirements and their perception of program benefits.	X		X	
2.5.06.	Actively promote the maturation of stakeholder requirements, for example, by providing detailed trade-off studies, feasibility studies, and virtual prototypes.				X
2.5.08.	Create effective channels for clarification of requirements (e.g., involving customer stakeholders in program teams).	X			
2.6.1.	Strive to minimize and streamline the burden of paperwork for external stakeholders by actively engaging them in the process and clearly articulating and aligning the benefit generated by each report.		X		
2.6.2.	Minimize and streamline the program-internal reporting for program activities and subprojects by optimizing the internal reporting requirements. Only require reports that are clearly necessary, and align reporting requirements to reduce redundant reporting.		X		
3.	**Lean Enablers to Optimize the Value Stream (Lean Principle 2)**				
3.01.02.	Promote reuse and sharing of program assets. Utilize standards, standard processes, modules of knowledge, technical standardization and platforms, and software libraries.	X			
3.03.2.	Explore the trade space and margins fully before focusing on a point decision and too small margins.				X
3.03.5.	Explore constraints and perform real trades before converging on a point design.				X
3.05.11.	Anticipate and plan to resolve as many downstream issues and risks as early as possible to prevent downstream problems.	X			
3.07.07.	Communicate to suppliers with crystal clarity all expectations, including the context and need, and all procedures and expectations for acceptance tests; and ensure the requirements are stable.	X			
4.	**Lean Enablers to Create Program Flow (Lean Principle 3)**				
4.03.3.	Ensure that the competency, technical knowledge, and other relevant domain knowledge of the program manager and the other key members of the program team are on par with the technical complexity of the program.	X			
4.05.	**Pursue collaborative and inclusive decision making that resolves the root causes of issues.**		X	X	
4.05.01.	If decisions are based on assumptions that are likely to change, keep track of those assumptions and adjust the decisions when they change.	X		X	
4.05.04.	Never delay a decision because you are not willing to take the responsibility or are afraid to discuss the underlying issues.			X	
4.05.10.	Proactively manage trade-offs and resolve conflicts of interest among stakeholders. Do not ignore or try to gloss them over.				X
4.07.	**Use efficient and effective communication and coordination with program team.**		X		
4.07.1.	Capture and absorb lessons learned from almost all programs.	X	X		
4.07.6.	Promote direct, informal, and face-to-face communication.			X	
4.08.3.	Promote design standardization with engineering checklists, standard architecture, modularization, busses, and platforms.	X			
4.08.4.	Promote process standardization in development, management, and manufacturing.	X			
4.09.	**Use Lean Thinking to promote smooth program flow.**	X	X		
4.09.01.	Use formal frequent comprehensive integrative events in addition to programmatic reviews: (a.) Question everything with multiple "whys"; (b.) Align process flow to decision flow; (c.) Resolve all issues as they occur in frequent integrative events; and (d.) Discuss tradeoffs and options.		X	X	X
4.09.06.	Use Lean tools to promote the flow of information and minimize handoffs. Implement small batch sizes of information, low information in inventory, low number of concurrent tasks per employee, small task times, wide communication bandwidth, standardization, work cells, and training.	X	X		
4.09.07.	Use minimum number of IT tools and make common wherever possible.	X	X		
4.09.08.	Minimize the number of the software revision updates (e.g., noncritical updates) of IT tools and centrally control the update releases to prevent information churning.	X	X		

4.09.09.	Adapt the IT tools to fit the people and process.	X	X		
4.09.10.	Avoid excessively complex and overly feature-rich IT tools. Tailor tools to program needs, not the other way around.	X	X		
4.10.1.	Make work progress visible and easy to understand to all, including external customer.	X			
4.10.4.	Develop a system that makes imperfections and delays visible to all.	X			
4.10.5.	Use traffic light system (green, yellow, red) to report task status visually (good, warning, critical) and make certain problems are not concealed.	X			
4.10.9.	Develop a snapshot/summary representation of the meaningful metrics (e.g., standard deck) to measure all phases of the project and program and make it available to all.		X		
5.	**Lean Enablers to Create Pull in the Program (Lean Principle 4)**				
5.1.2.	Promote the culture in which people pull knowledge as they need it and limit the supply of information to genuine users only.	X		X	
5.1.5.	Promote effective, real-time direct communication between each giver and receiver in the value flow, based on mutual trust and respect, and ensure both understand their mutual needs and expectations.			X	
6.	**Lean Enablers to Pursue Program Perfection (Lean Principle 5)**				
6.1.1.	Use existing program management standards, guidelines, and applicable organizational maturity models to your program`s best advantage.	X			
6.2.1.	Develop an integrated, long-term approach to implement Lean management practices in product portfolio planning and the entire enterprise.	X	X		
6.2.7.	Codify lessons learned and evaluate their effectiveness.	X			
6.2.8.	Look for new and innovative ways to work that add value.			X	
6.3.1.	Implement the basics of quality. Do not create, pass on, or accept defects.	X	X		
6.3.2.	Follow basic problem solving techniques (e.g., plan-do-check-act) and adopt a culture of stopping and permanently fixing problems when they occur.		X	X	
6.3.3.	Promote excellence under "normal" circumstances and reward proactive management of risks, instead of rewarding "hero" behavior in crisis situations.		X	X	
6.3.4.	Use and communicate failures as opportunities for learning emphasizing process and not people problems.		X	X	
6.3.5.	Treat any imperfection as an opportunity for immediate improvement and lesson to be learned, and practice frequent reviews of lessons learned.	X	X		
6.3.7.	Promote the idea that the program should incorporate continuous improvement in the organizational culture.	X		X	
6.4.	**Use lessons learned to make the next program better than the last.**	X			
6.4.1.	Create mechanisms to capture, communicate, and apply experience.	X	X		
6.4.2.	Clearly document context of "best practices" and "key learnings" in lessons learned to allow evaluation of appropriateness in new programs.	X			
6.4.3.	Create a process to regularly review, evaluate, and standardize lessons learned and prepare them for implementation.	X	X		
6.4.4.	Assign responsibility and accountability for reviewing, evaluating, and standardizing lessons learned and implement resulting change.	X	X		
6.4.5.	Insist on standardized root cause identification and process for implementing corrective action and related training.		X	X	
6.4.6.	Identify best practices through benchmarking and professional literature.			X	
6.5.	**Use change management effectively to continually and proactively align the program with unexpected changes in the program`s conduct and the environment.**	X			
6.7.01.	Develop a general program policy/guideline/framework that outlines expectations regarding communication, coordination, and collaboration.		X	X	
6.7.02.	Use concise one-page electronic forms (e.g., Toyota's A3 form) for standardized and efficient communication, rather than verbose, unstructured memos. Keep underlying data as backup in case it is requested by the receiver.	X	X		
6.7.03.	Similarly, use concise one-page electronic forms for efficient, real-time reporting of cross-functional and cross-organizational issues, for prompt resolution.		X		
6.7.04.	Develop a plan that implements the policy and ensures accountability within the entire program team in communications, coordination, and decision-making methods at the program beginning.		X	X	
6.7.07.	Publish instructions for artifact content and data storage, central capture versus local storage, and for paper versus electronic, balancing between excessive bureaucracy and the need for traceability.	X	X		
6.7.09.	Ensure timely and efficient access to centralized data.	X			
6.7.10.	Develop an effective body of knowledge that is easily accessible, historical, searchable, and shared by team and a knowledge management strategy to enable the sharing of data and information within the enterprise.	X			
6.8.01.	Utilize and reward bottom-up suggestions for solving employee-level problems.		X		
6.8.04.	Define a process that implements successful local improvements in other relevant parts of the program.	X			

4. Case Study: Kongsberg Automotive

4.1. About the company

Kongsberg Automotive (KA) is a global provider of development engineering and automotive products. The company is a parts manufacturer for different automotive applications such as seat comfort, suspension and driveline systems, clutch actuation and electronics for off highway and recreational vehicles. The technology development team consists of 450 R&D engineers spread around the world. KA holds a wide range of R&D disciplines and production methods. The company benefits from global resources arranged into regional centers of excellence, but at the same time maintaining the full range of R&D capabilities in North America, Europe and Asia.

With over 10.000 employees around the world, the company acquires large amounts of knowledge. The included knowledge is related to development and manufacturing. Today KAs knowledge base is mainly codified in Microsoft Office Excel. This knowledge base is stored in a Product Lifecycle Management (PLM) system from SAP. The PLM system stores all the documentation created during the lifecycle of the products. The Excel knowledge base consists of a tree structure with documents connected through lines showing their relation and legacy. The documents are mostly stored in a excel sheet or an A3.

The PLM system in KA contains massive amounts of information. The tree structure represents the knowledge value stream in symbiosis with product value stream by Kennedy, which is adapted by KA to present the knowledge flow within KA. For every project run in the company, the amount of possessed knowledge is rising. All of the projects provide more knowledge to the company. The problem is distributing the acquired knowledge to later projects through the PLM system.

All the documents in the PLM system are stored by using different programs from the Microsoft Office package or CAD programs. The PLM system does not convert, which is positive for maintaining integrity of files. However, because the files are stored in different file types, the user is not able to search within the text of the documents. The only search parameters are the file name, and the categorization stated by the file creator. Therefore, a search will come up with zero hits, if the search parameter deviates from the name or categorization of the specific file. The PLM system in KA has a folder structure, making it possible to seek manually through the folders. However, this is time consuming and often requires several attempts to find the right folder.

4.2. Transforming the product development process

The KBD journey of Kongsberg Automotive (KA) started within a limited group of people in the R&D department who got interested in KBD. The group quickly got support from the CEO of the company. The first step of methodology implementation required a change of mindset of the whole organization. It was considered to be critical to get support from the top to be successful, which was also the case. Michael Kennedy was invited to educate at the headquarter about KBD together with appropriate literature such as the book Managing to Learn[8] to increase knowledge about the mentor role. The role requires for them not to give answers to their team, but rather to ask the right set of questions and let them find the solution on their own. This is all part of the mindset change towards a KBD working methodology.

One part of the implementation is a change of documentation. The focus is to make A3's either for visualization and discussion or as documentation of knowledge gaps, customer interests or request for qualifications. The A3 is digital where text, pictures and graphs are the information formats.

As part of changing the company culture the Kongsberg Automotive Management System needed an update to show the new process. The descriptive figure of the system now has customer satisfaction in the middle surrounded by plan, do, check and act, explaining a learning cycle consisting of two LAMDA cycles. In the second ring the Knowledge Based Development process is defined.

4.3. Effects of introducing KBD

It is still early to see what effects on KA the process change has created. But one change to be observed is that people, according to the interviews, have an increased awareness of the knowledge value stream and not only the product value stream. In discussions and in their overall process plan, KA employees show maturity in applying and understanding underlying lean, KM and IT support systems. Still, they have a tendency to focus more on knowledge capture than knowledge reuse. An example is the attempt to integrate knowledge in the PLM system before concerning how and where the knowledge will be reused. KA however, has created a culture change in the company by starting to use A3s for documentation. A single A3 page to save the information about the subject is the basis for the process. This has created a more systematic approach to attack problems, all the way from understanding, through solution proposal to implementation.

5. Analysis

5.1. Mapping of Kongsberg Automotive's initiative to the Lean Enablers

Adapting a culture that focuses more on knowledge value stream and understanding the importance of continuous learning has proven successful across the company. Both the interviews and discussions around the LE show that there has been a change in this area. Much of LE on LAMDA culture shows that there is a complete agreement that this has been achieved, which is also supported by the observations. Some of the LEs that show full agreement are:

- Insist on standardized root cause identification and process for implementing corrective action and related training [6.4.5].
- When resolving issues, attack the problem, not the people [1.1.10].
- Follow the basic problem solving techniques and adopt a culture of stopping and permanently fixing problems when they occur [6.3.2].

Interviews show that even if KA focuses on and understands the value of knowledge, there are still things to do to improve knowledge reuse. One comment on this from the R&D department: "When you are in the project you know why some properties are the way they are, but after the project is finished it's hard to reach that information". This refers both to sharing knowledge between projects but also when the aftermarket seeks answers to a particular design. The study shows that there is no agreement on that "a process to regularly review, evaluate, and standardize lessons learned and prepare them for implementation [6.4.3]" is in place. In the beginning KA hoped that A3 was suitable to become their knowledge repository. But after some time they understood that A3 is supposed to work as a tool for discussion, sharing and workshops. Going through some of the A3s shows that a small amount of the A3 holds knowledge that should be transferred to Checksheets. Today this is only tested in a pilot stage at KA. Both interviews and LE show high agreement on achieving the A3 methods and benefits when used. Here are the related LE principles:

- Use concise one-page electronic forms (e.g. Toyota's A3 form) for standardized and efficient communication, rather than verbose, unstructured memo. Keep underlying data as backup in case it is requested by the receiver [6.7.02].
- Similarly, use concise one-page electronic forms for efficient, real-time reporting of cross-functional and cross-organizational issues, for prompt resolution [6.7.03].
- Insist on standardized root cause identification and process for implementing corrective action and related training [6.4.5].

Interviews and observations show that improvements still need to be made on the process and the quality of the A3s. Two comments are: "Lack of sufficient routines for making knowledge A3s when acquiring new knowledge" and "Misconception of expected quality of knowledge A3s" even if they agree to the above LEs.

Trade off graphs are rare at KA but the LEs that were categorized as measuring this aspect covered not specifically graphs but a wider space. The result was that the interviews and observations didn't really support each other. Nearly all LEs about trade-off showed full agreement on achievement even if trade-off curves were not broadly implemented. Examples of LE that were considered:

- Explore the trade space and margins fully before focusing on a point decision and too small margins [3.03.2].
- Actively promote the maturation of stakeholder requirements, for example, by providing detailed trade-off studies, feasibility studies, and virtual prototypes [2.5.06].

6. Discussion

Knowledge management should be fundamental in all systems engineering programs since technical knowledge forms the basis for creating new products and processes. Knowledge increases efficiency both on the company and product levels. The SECI framework explains knowledge creation through socialization, or interaction between individuals, and through internalization, or absorption of explicit knowledge[4]. A knowledge management system needs to have knowledge presented in several levels and forms that suit users with varying depths and ranges of knowledge[7]. KA's goal was to improve knowledge sharing in the organization and to support this the roll out was done.

KA has done a good job both in adapting tools and methods but also in trying to change the company's culture to accept the focus change from not only creating product but more of creating knowledge that results in products. Still there are things to consider such as the perspective of channels, content and learning in the knowledge value stream. Potential for improvement exists in the channels for ensuring that there is optimal flow without loss of knowledge within and across the system engineering programs. Necessary to consider during the system engineering programs is to focus on the content and proactively capture knowledge in a form that renders it readily usable and supports toward the optimal solution. By using improved system engineering approaches that facilitate accelerated learning cycles, knowledge will be gained faster and will increase the possibility for success by more often controlling the program in right direction.

The interviews, observations and LE studies show that to be even more successful with the KBD roll out, focus should be on balancing the knowledge life cycle and work more on giving the system engineer support to be more successful with their knowledge reuse. As expressed by LE the focus should be on '*Publish instructions for artifact content and data storage, central capture versus local storage, and for paper versus electronic, balancing between excessive bureaucracy and the need for traceability'* [6.7.07]. This means that there should be a balance between what "needs" to be captured and what would be "nice" to capture. The knowledge that is captured should also be codified in such a way that it supports reuse. This is guided by '*create process to regularly review, evaluate, and standardize lessons learned and prepare them for implementation'* [6.4.3].

The analysis in this study shows that the LEs linked to the four tools and methods work well for the LAMDA culture, A3 and likely with CS but do not correlate fully when it comes to trade-off curves. The LE mapped for trade-off curves measures a broad perspective that can be misleading.

During the implementation of the different tools and methods, observations and parallel interviews have identified especially four success elements that are seen as more important for the success of the KBD roll out. Before changing the process, the benefits that it strives to achieve need to be communicated and also why the process is being changed what activities it entails and when they are scheduled.

LAMDA as a culture is expressed as a method but is also identified as a success element for all the implementations. People need to be aware of the importance of learning and that knowledge is extremely valuable so it should be easy to access and use. A change in is not easy to achieve but lays the ground for future changes to the process. The goal with this culture change is also to drive employees to naturally share their knowledge.

Before other tools and methods are chosen and implemented it's necessary to **understand and outline organizational needs** so the implementation procedure fits. Here, both internal and external stakeholders are affected, which needs to be taken into consideration. Identifying these needs will later on be of value to make sure that commitment from employees and management stays strong or increases.

Organizational commitment is necessary when adapting the process to a new way of working which is a change both physically and in mindset for all parts. Leading by example is a valuable part where initiatives like "train the trainer" increase commitment from employees and also create a deeper understanding. An assessment of the current status like the one performed at KA increase the commitment from the top and can be used to create the future roadmap.

When commitment, understanding of organizational needs and the culture is in place a **pilot tool** can be implemented. To gain an important success from beginning the implementation should start with a simple tool that is linked back to the needs. A successful pilot will also decrease the risk of "not-invented-here" syndrome. During pilot stage the team needs to test and make necessary adjustments to increase the performance. But each proposed change needs to be clarified and understood together with stated benefits before they are implemented to make sure that none of the existing benefits come to harm. Continuous improvements are necessary but without forgetting that a standardized product must be in order to secure the basic functionality. The pilot test serves as an example for the rest of the organization and provides necessary lessons for the full implementation.

7. Conclusion

KA has adapted methods and tools from KBD to increase quality of their products and decrease rework and repetitive errors. During the time a transformation in mindset has been seen, people have become aware of the knowledge value stream and instead of only focusing on building better products, learning has been added as a part of the system engineering approach and knowledge is asked for as a separate deliverable from the managers. The paper clarifies how an organization can work with different KBD tools and methods to fulfill different needs over the knowledge lifecycle. The case shows the need to focus more on the reuse perspective than previously done.

Creating knowledge based on truly understanding the problem with the support from A3 and LAMDA has been successful and satisfactory. However, there is still no standardization of how and when an A3 should be created.

Today A3, guidelines and standard documents do not support knowledge reuse in a satisfactory way. The conclusion from this study is that the knowledge is not adapted to fit the user, and the packaging does not increase accessibility and availability. Checksheets is the next tool that KA is aiming to implement and an initial prototype shows that it supports the engineers better for reuse of existing knowledge, which is also described in theory.

The KBD process needs to be further validated and developed. The result so far is that the company will continue to explore all four tools and methods in different system engineering groups and especially Checksheets will subsequently be analyzed in order to see if it supports and increases knowledge reuse. During implementation of the different tools, four success elements have been identified: the culture change, understanding organizational needs, building organizational commitment and pilot tool.

Acknowledgements

This study was carried out at the Wingquist Laboratory VINN Excellence Centre within the Area of Advance Production at Chalmers, and supported by the Swedish Governmental Agency for Innovation Systems (VINNOVA).

The research has been funded by the research council of Norway and Kongsberg Automotive as a part of the KBD project.

References

1. Kennedy Michael 2008. *Ready, set, dominate: Implement toyota's set-based learning for developing products and nobody can catch you,* Richmond, VA, Oaklea Press.
2. Mascitelli Ronald 2007. *The lean product development guidebook: Everything your design team needs to improve efficiency and slash time-to-market,* Northridge, CA, Technology Perspectives.
3. Morgan James M & Liker Jeffrey K 2006. *The toyota product development system,* Seattle, WA, Productivity Press.
4. Nonaka Ikujiro 1994. A dynamic theory of organizational knowledge creation. *Organization Science,* **5**, 14-37.
5. Oehmen Josef, Oppenheim Bohdan W, Secor Deborah, Norman Eric, Rebentisch Eric, Sopko Joseph A, Steuber Marc, Dove Rick, Moghaddam Kambiz & Mcneal Steve 2012. *The guide to lean enablers for managing engineering programs,* Cambridge, MA, Joint MIT-PMI-INCOSE Community of Practice on Lean in Program Management.
6. Oppenheim Bohdan W, Murman Earll M & Secor Deborah A 2011. Lean enablers for systems engineering. *Systems Engineering,* **14**, 29-55.
7. Raudberget Dag & Bjursell Cecilia 2014. A3 reports for knowledge codification, transfer and creation in research and development organisations. *International Journal of Product Development,* **19**, 413-431.
8. Shook John 2008. *Managing to learn: Using the a3 management process to solve problems, gain agreement, mentor and lead,* Cambridge, MA, Lean Enterprise Institute.
9. Sobek Ii Durward K & Smalley Art 2011. *Understanding a3 thinking: A critical component of toyota's pdca management system,* Boca Raton, CRC Press.
10. Stenholm D & Landahl J. Knowledge management life cycle: An individual's perspective. DS 77: Proceedings of the DESIGN 2014 13th International Design Conference, 2014.